1. **Постановка задачи**

В ходе данной лабораторной работы необходимо реализовать несколько архитектур полностью связанных нейронных сетей для решения практической задачи компьютерного зрения, используя одну из библиотек глубокого обучения.

Выполнение лабораторной работы подразумевает выполнение следующих задач:

1. Выбор библиотеки установка и проверка корректности ее работы на тестовом примере сети для решения задачи классификации рукописных цифр.
2. Выбор практической задачи компьютерного зрения.
3. Разработка программ/скриптов для подготовки тренировочных и тестовых данных.
4. Разработка нескольких архитектур полностью связанных нейронных сетей с различным количеством слоев и видами функций активации.
5. Обучение и тестирование разработанных глубоких моделей.
6. **Тренировочные и тестовые наборы данных**

В качестве набора данных были использованы фотографии с аватарок пользователей социальной сети Instagram. Выборка состоит из 40 976 изображений размера 320×320. Выборка разбивалась на тренировочную и тестовую в соотношении 80:20. Предварительно проводилась разметка данных – для каждого изображения задавался пол человека на фотографии (0 – мужчина, 1 – женщина, 2 – не определено).

1. **Разработанные программы/скрипты**

Проект состоит из двух основных частей:

1. Система для разворачивания датасета

* unzip\_dataset.py – разархивирует данные и вызывает конвертер;
* [tfrecords\_converter.py](https://github.com/mabean/DeepLearning/blob/master/data/tfrecords_converter.py) – создает из полученных данных tfrecords;
* [tfrecords\_reader.py](https://github.com/mabean/DeepLearning/blob/master/data/tfrecords_converter.py) – читает данные пачками из tfrecords.

1. Реализация нейросети

* [bespalov\_twolayer\_nn\_mnist\_example.py](https://github.com/mabean/DeepLearning/blob/master/Lab2/src/bespalov_twolayer_nn_mnist_example.py), [tensorflow\_example.py](https://github.com/mabean/DeepLearning/blob/master/Lab2/src/tensorflow_example.py) – тестовый пример сети для решения задачи классификации рукописных цифр;
* [flnn.py](https://github.com/mabean/DeepLearning/blob/master/Lab2/src/flnn.py) – реализация нейронной сети для трехканального изображения;
* [flnn\_one\_channel.py](https://github.com/mabean/DeepLearning/blob/master/Lab2/src/flnn_one_channel.py) – реализация нейронной сети для одноканального изображения (используется только компонента яркости.

1. **Тестовые конфигурации сетей и результаты эксперимента**

Для решения задачи были реализованны две сети различных конфигураций:

* Трехслойная нейронная сеть для трехканального изображения:

|  |  |  |  |
| --- | --- | --- | --- |
| Номер эксперимента | Количество нейронов на 1-м скрытом слое | Количество нейронов на 2-м скрытом слое | Точность на тестовом наборе (%) |
| 5 | 700 | 200 | 0.2048 |
| 4 | 800 | 400 | 0.2477 |
| 2 | 900 | 500 | 0.2517 |
| 1 | 800 | 300 | 0.3184 |
| 3 | 1000 | 300 | 0.3223 |

* Трехслойная нейронная сеть для одноканального изображения:

|  |  |  |  |
| --- | --- | --- | --- |
| Номер эксперимента | Количество нейронов на 1-м скрытом слое | Количество нейронов на 2-м скрытом слое | Точность на тестовом наборе (%) |
| 4 | 1000 | 300 | 0.2794 |
| 1 | 1000 | 500 | 0.2843 |
| 6 | 1100 | 400 | 0.3256 |
| 2 | 1300 | 300 | 0.3403 |
| 7 | 1100 | 300 | 0.3701 |
| 5 | 1200 | 300 | 0.3924 |
| 3 | 1200 | 400 | 0.4246 |